**Part 1)**

Link to Notebook: <https://github.com/HamzaKazmi/ShopifyChallange/blob/master/Shopify%20Data%20Science%20Challange.ipynb>

**Answers:**

**a)**

Outliers are likely skewing the result and giving a high average. These can be wrong entries, or some orders can be very large. Firstly, to get a better result when data is skewed is to use median instead of mean. Secondly, we can clean the data and get rid of any outliers which will give us a more accurate mean.

**b)**

Short Answer:

The data is skewed by two different shops:

Shop 42 is the only shop with bulk orders of 2000 sneakers totalling to order amount of 70400.

Shop 78 is the only shop with price/sneaker of 25725, which is very high in comparison to other shops as the median for price/sneaker is 153.

A better metric for when data is skewed is to use the median instead of the mean.

However, I also explain what shops are skewing the results, and then I remove those orders and recalculate the AOV. I found that after outlier removal the new mean is 304 with a median of 284.

**My thought process:**

Firstly, I check for duplicate, nan, and outlier values. There are no duplicate order entries nor there are any nan values. I then check for any values that might be too small. I look at the minimum order amount and it is 90. This tells me there are no outliers where price is too small. Then I check for order amounts larger than 5000 and discover that only shop 42 and 78 have order amounts over 5000 and maximum order amount is 70400. Shops 42 and 78 order amounts are very high for an order and can be considered an outlier as the mean is 3145 and median is 284. This value is also what is likely making the mean so high in comparison to the median.

70400 can be a wrong entry or it can be because the order is very large. I then create a column for the price/sneaker by dividing order amount with the total items. I use this column to check if price per sneaker is consistent for the shop which has order with amount 70400. After checking it shows that the price/sneaker is accurate and that the total number of items being sold is very large at 2000. This is likely some bulk orders specific to this shop as the second highest number of items in an order is 8. Therefore, I then remove these bulk orders from the dataset. Next, I also found that shop 78 sells a sneaker for 25725 which is also very high in comparison to what other shops are selling their sneakers for. The mean and median for a sneaker are 387 and 153. Therefore, the very expensive sneaker price for shop 78 are also sewing the results. Therefore, I remove shop 78 as well.

Therefore, after removing bulk orders from shop 42 and removing shop 78 which sells expensive sneakers, the average order value is 302 with a median of 284.

**c)**

Median: 284

AOV after outlier removal: 304

Q1)

If we know that Speedy Express is shipper with ID 1:

SELECT Count(ShipperID)

FROM [Orders]

where ShipperID == 1

If there are thousands of shippers and Speedy Express ID is unknown:

SELECT Count(Orders.ShipperID)

FROM Orders

INNER JOIN (SELECT \* FROM Shippers where ShipperName like 'Speedy Express') Shippers

ON Orders.ShipperID == Shippers.ShipperID

**Answer – 54**

Q2)

SELECT Employees.LastName

FROM Employees

INNER JOIN(SELECT EmployeeID, COUNT(OrderID) FROM Orders GROUP BY EmployeeID ORDER BY COUNT(OrderID) DESC LIMIT 1) HighestOrders

ON Employees.EmployeeID = HighestOrders.EmployeeID

**Answer – Peacock with 40 orders**

Q3)

SELECT Products.ProductName

FROM Products

INNER JOIN

(SELECT x.ProductID,SUM(x.Quantity)

FROM Customers

INNER JOIN

(SELECT OrderDetails.ProductID,OrderDetails.Quantity,Orders.CustomerID

FROM OrderDetails

INNER JOIN Orders ON OrderDetails.OrderID == Orders.OrderID) x

ON Customers.CustomerID == x.CustomerID AND Customers.Country like 'Germany'

GROUP BY ProductID

ORDER BY SUM(x.Quantity) DESC LIMIT 1) y

ON Products.ProductID == y.ProductID

**Answer – Boston Crab Meat**

Thank you for the consideration!